Computer Systems Science & Engineering
DOI: 10.32604/csse.2023.028400
Article

Tech Science Press

Route Planning for Autonomous Transmission of Large Sport Utility Vehicle

V. A. Vijayakumar[*], J. Shanthini, S. Karthik and K. Srihari

SNS College of Technology, Coimbatore, 641035, Tamilnadu, India
*Corresponding Author: V. A. Vijayakumar. Email: vijaykumarphd12@gmail.com

Received: 09 February 2022; Accepted: 12 April 2022

Abstract: The autonomous driving aims at ensuring the vehicle to effectively sense the environment and use proper strategies to navigate the vehicle without the interventions of humans. Hence, there exist a prediction of the background scenes and that leads to discontinuity between the predicted and planned outputs. An optimal prediction engine is required that suitably reads the background objects and make optimal decisions. In this paper, the author(s) develop an autonomous model for vehicle driving using ensemble model for large Sport Utility Vehicles (SUVs) that uses three different modules involving (a) recognition model, (b) planning model and (c) prediction model. The study develops a direct realization method for an autonomous vehicle driving. The direct realization method is designed as a behavioral model that incorporates three different modules to ensure optimal autonomous driving. The behavioral model includes recognition, planning and prediction modules that regulates the input trajectory processing of input video datasets. A deep learning algorithm is used in the proposed approach that helps in the classification of known or unknown objects along the line of sight. This model is compared with conventional deep learning classifiers in terms of recall rate and root mean square error (RMSE) to estimate its efficacy. Simulation results on different traffic environment shows that the Ensemble Convolutional Network Reinforcement Learning (E-CNN-RL) offers increased accuracy of 95.45%, reduced RMSE and increased recall rate than existing Ensemble Convolutional Neural Networks (CNN) and Ensemble Stacked CNN.

Keywords: Artificial intelligence; information system; security and privacy; fuzzy modelling; deep neural networks; machine learning; reinforcement learning; CNN; SUV

1 Introduction

In the recent past, autonomous vehicles improved its efficiency and safety potential [1,2]. The operation of vehicles in dynamic environments with coordination, control and planning of paths [3–6]. The generalization capacity requires suitable operation in a prompt way, in the complex and terrain environment, to achieve human reliability and security. In this respect, informed and uninformed prediction model decisions should demand an accurate view [7].

To achieve the minimal error rate, several computer vision systems are used up to now on autonomous driving [8–10]. Most systems fail to reach reasonable error rates because the computer visual navigation system makes incorrect decisions [11,12]. The approaches to decision making, control and perception modules combined in recent times have produced promising results [2].

Problem Definition: The main problem associated with an autonomous driving using vehicle policy is the collection of larger collection of data for training the required model and an inappropriate mapping of the input with respect to the action output derived [13,14]. In order to reduce the limitation between the input and the output i.e., monitoring and controlling of an autonomous vehicle, an efficient and a faster processing model is hence required [15,16]. This optimally can parallelize the input operations by monitoring effectively the line-of-sight condition and provides the action output in a near optimal time [17]. Such a model should reduce the time consumption and it should reduce the intervention of human during the model training [18].

However, most of the deep learning system fails in predicting the environment after suitable recognition or classification of objects in an environment. This is specifically true in case of video recognition systems, where the deep learning system has to incur the objects along the trajectories of the path in motion.

Objectives: The application of deep learning algorithm with ensemble-based model has motivated the present study to classify the objects in faster and accurate way. Such that the prediction and planning of paths for autonomous SUVs can enable faster transmission of vehicle that is collision resistant. In this paper, a method on direct realization is developed on an autonomous vehicle driving is carried out using a hybrid ensemble model.

- To incorporate recognition, planning and prediction modules to regulate the input trajectory of video datasets.
- To aid the process of classification an ensemble Convolutional Neural Network (CNN) is used for the classification of objects along the video trajectories along the line of sight.
- To aim at optimal prediction for path planning using Reinforcement Learning (RL).

The main contribution involves the following:

- Authors developed an autonomous driving that deploys an ensemble classifier involving CNN and RL.
- An Ensemble CNN (E-CNN) model is developed using deep learning that is trained with the driving patterns for possible prediction of objects and enables smooth flow of driving based on the input trajectories and it ensures the vehicle to be driven in an automated way.
- The E-CNN is designed as a base classifier, where the ensemble system involves other layers namely: stacking layer and a meta-classifier. The stacking ensemble layer collects the features in optimal way and sends to the RL for final classification. The meta-classifier uses RL classifiers in parallelized way that ensures the processing of extracted input real time features from the datasets in faster way.
- The execution of an ensemble learning classifier is carried out on a path planning ensemble model for a careful autonomous driving specifically on a terrain and on a rugged surface.
- The performance of the ensemble classifier is carried out on a rugged and terrain areas in terms of classification accuracy on objects, recall rate and Root Mean Square Error (RMSE).

The outline of the paper is given below: Section 2 provides the related works with existing models used in improving the automated detection of road accidents. Section 3 details the Ensemble deep learning model with detailed steps on how the E-CNN-RL works. Section 4 evaluates the ensemble prediction model with a large video dataset over different landscapes. Section 5 concludes the entire work.

2 Related Works

Complex planning and decision-making, where definite driving performance have remained a barrier, has led to the integration of machine learning and deep learning modules. Using deep learning models

[19–25] has been shown to be successful in various object tracking applications. However, this comes with the requirement for significant supervised training, which will increase the complexity of the tracking system [26–29].

Data production without supervision would impact a model trained with redundant training sets. In order to recognise the environment in which predictable and unpredictable things are regarded, we must limit the data required to train the reinforcement learning model [30–39] to a set of data points.

3 Methods

In this section, a framework or a model is designed with ensemble model that incorporates recognition and prediction, and planning. The selection of algorithms is considered in a careful way such that the recognition and prediction of optimal trajectories is carried out in accurate way. The architecture in Fig. 1 shows the autonomous driving system.

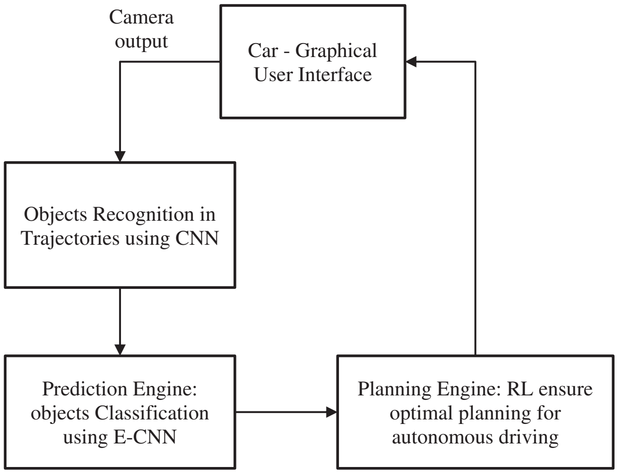

Camera output
Car - Graphical User Interface
Objects Recognition in Trajectories using CNN
Prediction Engine: objects Classification using E-CNN
Planning Engine: RL ensure optimal planning for autonomous driving

Figure 1: Framework of autonomous driving

It is not sufficient for an autonomous prediction model to recognize its environment while the vehicle is moving at high speed. Hence it is essential to develop an internal model, which can predict the future environmental conditions. The intelligence level with the hybrid ensemble learning model operates the car on terrain and rugged surfaces autonomously.

The Ensemble-Convolutional Neural Network-Reinforcement Learning (E-CNN-RL) classifier is equipped with recognition, prediction and planning.

Step 1: The recognition module uses a camera to capture the path

Step 2: The prediction module finds the objects.

a) It uses a CNN classifier [36–39] that helps in predicting the classes

Step 3: Stacked layer encounter weights of multi-labels obtained from the base CNN classifier.

Step 4: The RNN planning module reacts to the environmental changes.

a) The data from the fully convolutional layer is further sent to the reinforcement model that tends to process each action of the CNN with its actor-critic model (Fig. 2). If the retrieved instances are positive, the RL rewards it and if the instances obtained does not match with the query raised, then RL penalizes it. In this manner, the overall process of RL works. It acts as a feedback model of the CNN in automated detection.

b) Estimate the objective function in the planning models to achieves high quality prediction output

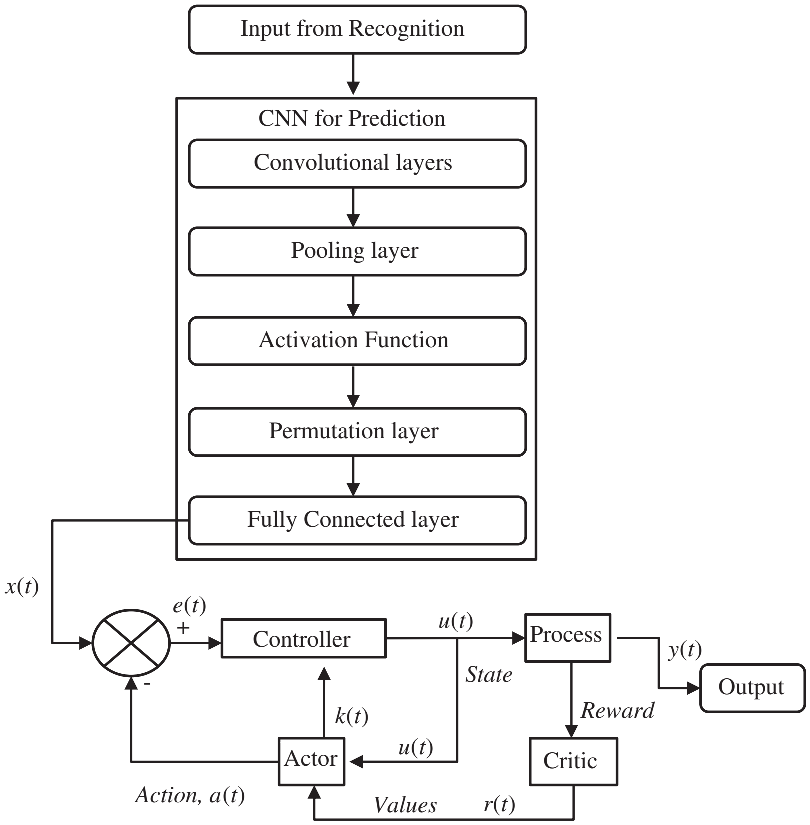


Figure 2: Automated detection module with RL

3.1 Recognition

In the recognition phase, we use a high-speed camera to collect all of the items in the surrounding settings for training and testing. In the algorithm, redundant frames, such as items that are not on the trajectories, are ignored. A small percentage of residual data, which includes all objects, is used as an input to CNN.

3.2 Prediction Using E-Cnn

Increasing the number of convolutional layers yields a more defined movement. Moving objects are detected by the convolution layer after training on ImageNet. Also, it removes static objects in the foreground of the incoming video frames.

- *Convolutional layers:* The linear processing operation performed by convolutional layers involves multiplying each video input data array weight by the input video data array. Translation invariance is utilised to locate the saliency of the input image, also known as translation invariance. A two-dimensional feature map is formed since filters are used several times in the input array.
- *Pooling layers* in CNN is capable of performing down-sampling process, which is of non-linear one. The most commonly used operation is the max pooling, where it splits the input into non-overlapping

sets and the maximum sets are generated as output from each group. This helps to reduce the spatial representations, parameters, overfitting and computational complexity of the convolutional network.

- *Activation function* used most commonly in the study us rectified linear unit function $f(x) = \max(0, x)$.
- *Permutation layers*: After the convolution layer, the reference objects are listed in order that the permutation layers validates the object being categorised by listing them in permutation order. Using the two permutations in place of the distance function calculates the correlation. The ground truth and the reference item can be compared to each other. Relationships usually behave in a manner similar to that exhibited by distance.
- *Fully Connected Layers* consists of neurons, which are fully connected with entire previous layer activations and it is applied after the convolutional and permutation and pooling layers.

The ensemble base layer in the present layer consists of multiple CNNs that generates multiple labels based on supervised learning sequence. The multiple labels are then process by the stacked ensemble layer based on its weighted function.

3.3 Stacked Ensemble

Stacked ensemble model is an advanced ensemble model that is designed to improve the precision/accuracy of the prediction. The present study uses this stacked ensemble model that uses ensemble learning algorithm. The meta-learning stacked ensemble is the best suited module that learns to combine well the predictions from two or more CNN base classifier. The stacking in the ensemble algorithm harnesses the ability of CNN classification and predicts using RL how they will work better than any single ensemble model.

The prediction from the multiple CNN classifier with relevance to road trajectories offers accurate autonomous driving by the vehicle without the intervention of the driver. The stacking using multi-label classifier involves the application of binary relevance and considers the prediction of CNN as a meta-level feature for final planning model. This ensures correlation among the supervised/labeled data in the planning model or at the meta-level. The problem of overfitting is avoided using cross validation of the planning classifier. Here, the data is split into disjoint parts that generates base classifier N times using $N - 1$ partitions each time for training and the remaining utility is used for planning via RL. With such varied feature space from individual CNN classifier, the ensemble classifier is diversified. The stacked ensemble learning considers exploits the local and global pairwise label correlation for prediction and planning.

The weighted stacked ensemble tends to reduce the distance between the predicted score and the target vector (representing the ground truth information in the label space). This is represented as the minimum Euclidean linear least-square problem, which is given as in Eq. (1):

$$\min_{w} \|y - p_w\|_2^2 \tag{1}$$

where, y is the target vector, w is the weight vector and p is the prediction score matrix.

In order of improving the performance, various data types at meta-level is taken into account that can either be discrete or continuous values. The introduction of irrelevant or non-labelled classified objects from the CNN can be introduced in RL engine. In such cases, the information is uncorrelated with respect to the prediction by CNN engine/base classifier. Hence, the classification performance degrades with the addition of non-labelled information and noises. In such cases, the present study introduces the weights based on the confidence scores obtained from the various CNN base classifiers with respect to varying labels.

3.4 Planning Using Rl

Reinforcement Learning (RL) is an approach that learn the environment through interactions for increasing the cumulative reward signals. An agent for learning interacts actively with an environment at all states (u). Action (a) is chosen based on a decision from a set of actions. Each action i.e., feature extractions are validated and that results in scalar reward i.e., reward signal (R). The Markov decision process makes the sequential decision based on conditionally independent states and actions from past states and actions. Such considerations are made based on Markov assumptions and it learns the optimal policy that helps in maximizing both the immediate and future reward. A dynamic programming is utilized for the computation of optimal function, where the agents predicts whether the feature extraction carried out on an image frame in an online video stream is valid or not. The validation is carried out based on the training inputs or the target landmark from the input datasets. A similarity comparison is made by the agents on predicted static and dynamic features, and the target features. It the action is valid, then the agent rewards CNN output and if the action in invalid i.e., the feature extracted are not precise, then the agent penalizes the CNN. The algorithm for RL is thus given below:

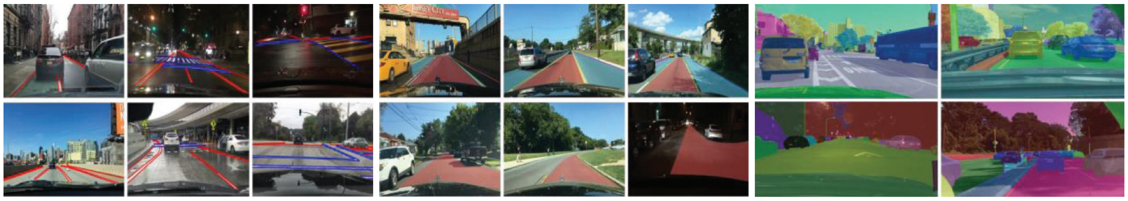
3.5 Algorithm 1: RL

```
    Initialize state 0: $u_0(t)$, action at state 0: $a_0(t)$, discount factor ($\gamma$) and learning rate ($\alpha$)
    Repeat $t = 0$
    Initialize the state $u(t)$
    Choose an action $a(t)$ from the state $u(t)$ through the Markov decision process policy from RL
    Assume $\varepsilon$ as a greedy function
    Repeat at each iteration
    Take action $a(t)$ based on the observation from observe $u(t+1) \rightarrow$ reward $r(t)$ or penalty $p(t)$
    Consider the next action $a(t+1)$ from next state $u(t+1)$ through the Markov decision process from RL
    Assume $\varepsilon$ as a greedy function
$RL(u(t),\ a(t)) \leftarrow Q(u(t),\ a(t)) + \alpha[r(t) + \gamma Q(u(t+1),\ a(t+1)) - Q(u(t),\ a(t))]$
$u(t) \leftarrow u(t+1);\ a(t) \leftarrow a(t+1);\ r(t) \leftarrow f(y(t))$
    Repeat the action until entire state $u(t)$ terminates
```

The weighted labels obtained from the sequence or video frame of base and then stacked layer is used to train the RL to learn the model for path planning that steers the car to move forward, sideward (left/right), backward, to slow, to speed, and to incline the car at certain degrees to move without any obstacle at its foreground. The decision is based on the weighted labelled sequence from each base classifier that sets to consider optimal detection of objects in motion (both temporal and spatial objects). The decision of autonomous driving is initiated with repeated training and marking each action of the RL with reward/ penalty actions. The rules for driving based on the obstacle is designed autonomously by Fuzzy Logic controller. If the car is supposed to move in proper path, the RL is assigned with reward vector and vice versa. In this way, the RL learns the entire planning process and predefined actions based on the labelled sequence is embed during training process and finally the RL is set for testing and validation.

4 Results and Discussions

In this section, the validation of the E-CNN-RL model is presented. The entire modelling is coded using Python Scripts in a Pytorch framework. In Figs. 3a–3c, the study uses BDD100K [30] dataset: A Large-scale Diverse Driving Video Database for training and validation that provides the details of four major objects that includes its diversity, large-scale data collection, captured on a street and temporal information. The study is evaluated in terms of Recall Rate, RMSE and Accuracy.



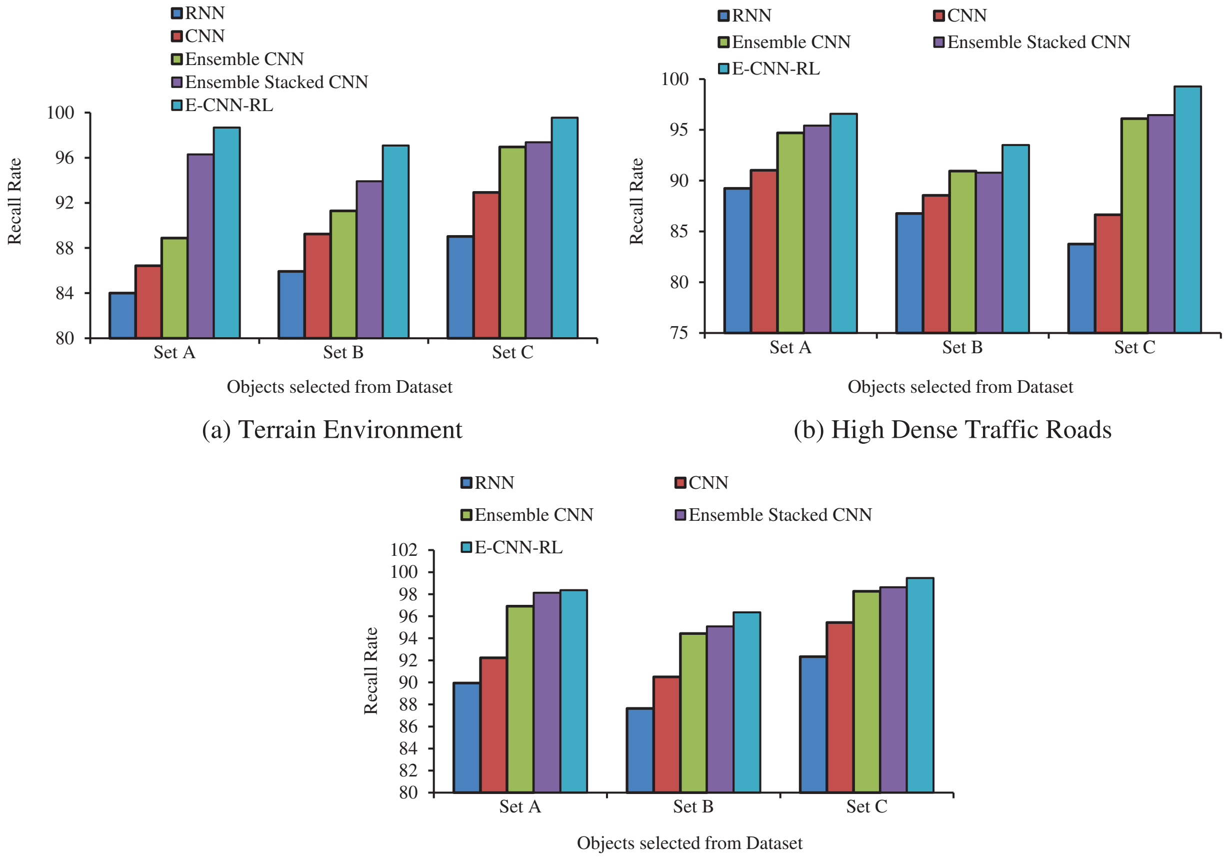
(a) Lane Markings (b) Drivable Areas (c) Segmented Lanes

Figure 3: Supervised training data from BDD100K dataset

The model is trained on various environment and this is provided below:

- ***Set A-Terrain***: It contains various soil conditions, slopes, micro-relief, and meteorological conditions.
- ***Set B-Traffic (High dense)***: Medium and heavy traffic comes under the category of high dense roads. It is otherwise defined as when there many vehicles on the road and the vehicles moves slower, the traffic is considered as high dense. This category does not fall under terrain environment.
- ***Set C-Free roads (less dense)***: Low heavy traffic comes under the category of high dense roads. It is otherwise defined as when there are not many vehicles on the road and the vehicles can move faster, the traffic is considered as less dense. This category does not fall under terrain environment.

Figs. 4a–4c shows the testing recall results in terrain environment (Fig. 4a), dense traffic roads (Fig. 4b) and less traffic roads (Fig. 4c) after 1000 runs of training. The results show that the E-CNN-RL attains higher recall rate than other classifiers.

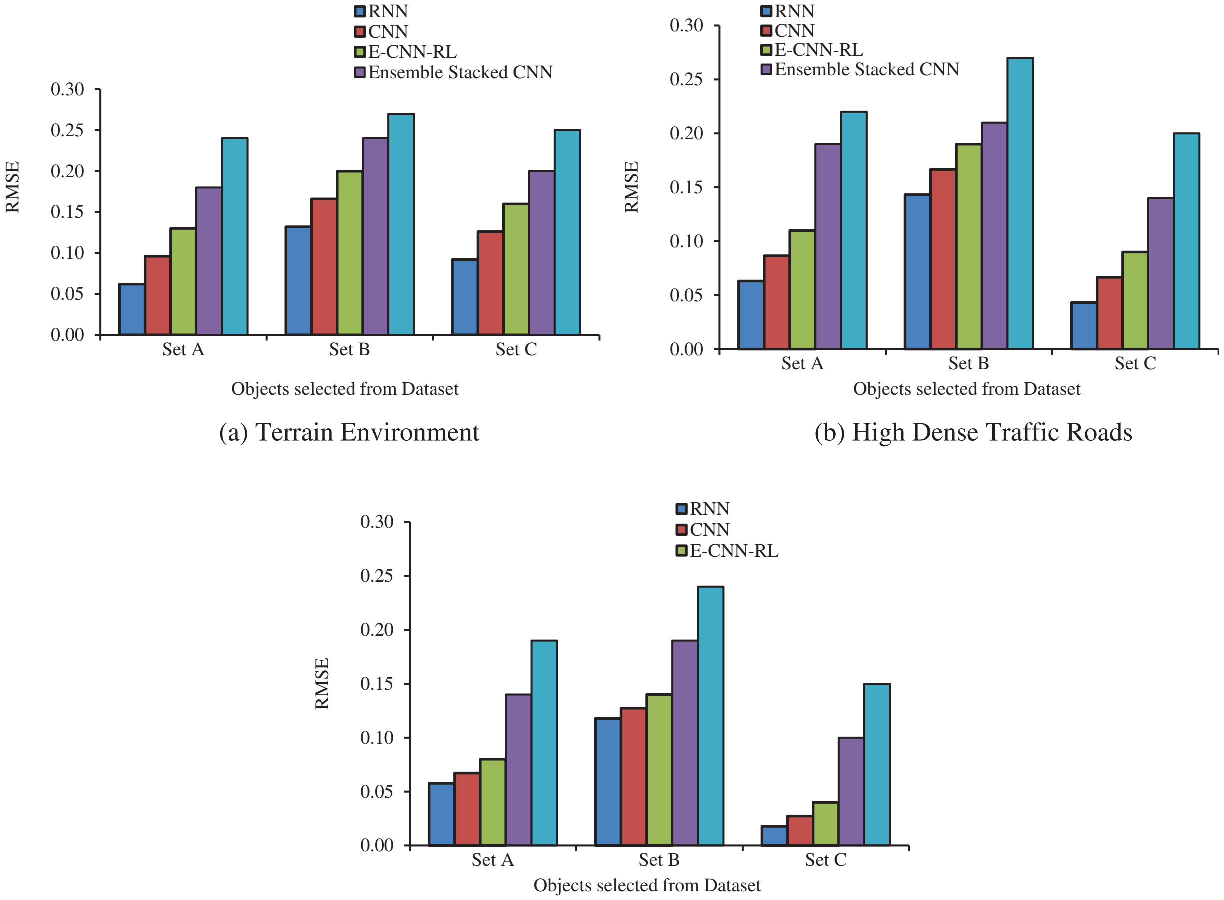


(a) Terrain Environment (b) High Dense Traffic Roads

(c) Less Dense Traffic Roads

Figure 4: Results of recall rate on various traffic types

Figs. 5a–5c shows the testing RMSE results in terrain environment (Fig. 5a), dense traffic roads (Fig. 5b) and less traffic roads (Fig. 5c) after 1000 runs of training. The results show that the E-CNN-RL attains reduced RMSE rate than other classifiers. The evaluation shows that the C has increased features with reduced RMSE rate, and shows optimal results in E-CNN-RL than existing classifiers. The results show that the E-CNN-RL has reduced RMSE rate than the E-CNN and stacked E-CNN classifier and further the RMSE for E-CNN-RL has reduced RMSE rate in C than A and B.



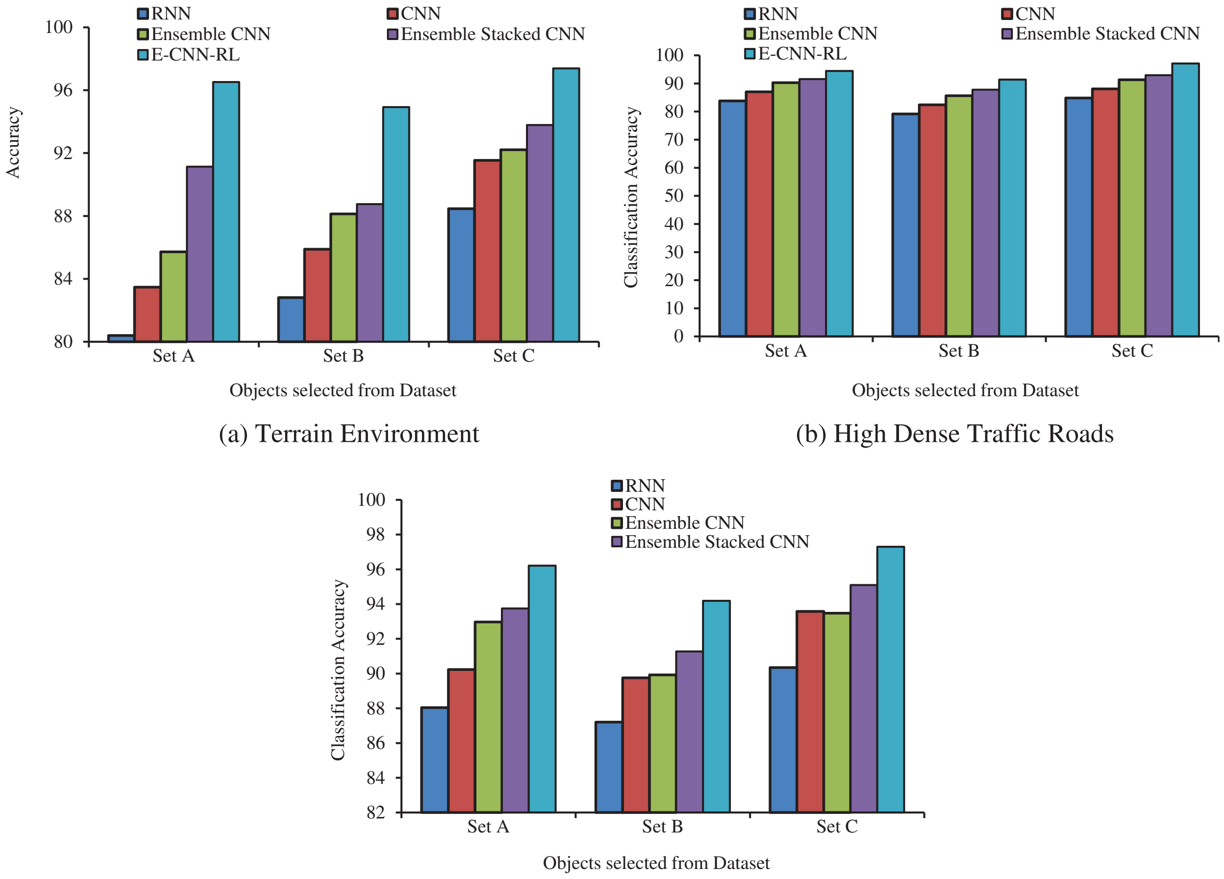
(a) Terrain Environment

(b) High Dense Traffic Roads

(c) Less traffic Roads

Figure 5: RMSE results on various traffic types

Figs. 6a–6c shows the testing accuracy results in terrain environment (Fig. 6a), dense traffic roads (Fig. 6b) and less traffic roads (Fig. 6c) after 1000 training runs. The results show that the E-CNN-RL attains reduced RMSE rate than other classifiers. The increase in performance using E-CNN-RL is the results of the reduced redundant items at the base classifier and with CNN at object recognition and prediction and repeated reinforcement learning at the planning stages. As per the labels of various video frame sequences, the stacked layer trains RL and plans the path, where the objects are found and possible the car is made to move away from the objects such that no collision exist with other vehicles in different environmental conditions. The objects at different surface and road conditions are learnt during the training and possibly the testing phase executes an automated driving pattern.



(a) Terrain Environment

(b) High Dense Traffic Roads

(c) Less Dense Traffic Roads

Figure 6: Results of classification accuracy on various traffic types

5 Conclusions

In this paper, a hybrid E-CNN and RL model adopts well with the recognition of objects along the input trajectory and making optimal decisions of vehicle moving in rugged and terrain environment. The CNN offers improved detection of objects with its state and action mechanism and the error rate of CNN reduces with increasing iterations. The training and testing of the ensemble meta-classifier i.e., RL offer optimal planning and predictions based on the improved E-CNN object detection in fast moving cars. The experiments show that with reduced speed, the accuracy is more, however, the objective of achieving high accuracy during high-speed transmission is achieved using E-CNN object detection and RL path planning in rugged and terrain environment. The validation confirms the optimal path planning with accident-free driving. The supervised E-CNN learning by the base classifier assists the entire ensemble approach to offer optimal decisions on path planning. The faster classification of objects along the trajectories i.e., both labelled and non-labelled objects with multiple base class provides effective path planning than previous systems.

The study can further be improved by considering the video saliency associated with object detection that involves both static and dynamic video saliency in future prediction systems. This can especially be tested on off-road vehicle for torsio-elastic suspension applied to front, rear and both axles [33]. This system can be extended with deep learning modules on base/meta classifier to enrich the transmission of hybrid vehicles in case of component sizing optimization [31] and wheel interaction measurement [32].

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] L. M. Clements and K. M. Kockelman, "Economic effects of automated vehicles," *Transportation Research Record*, vol. 2606, no. 1, pp. 106–114, 2017.

[2] W. Schwarting, J. Alonso Mora and D. Rus, "Planning and decision-making for autonomous vehicles," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, pp. 187–210, 2018.

[3] R. A. Raja and N. V. Kousik, "Analyses on artificial intelligence framework to detect crime pattern," *Intelligent Data Analytics for Terror Threat Prediction: Architectures, Methodologies, Techniques and Applications*, Wiley, pp. 119–132, 2021.

[4] M. G. Mohanan and A. Salgoankar, "A survey of robotic motion planning in dynamic environments," *Robotics and Autonomous Systems*, vol. 100, pp. 171–185, 2018.

[5] S. D. Pendleton, H. Andersen, X. Du and X. Shen, "Perception, planning, control, and coordination for autonomous vehicles," *Machines*, vol. 5, no. 1, pp. 1–6, 2017.

[6] Y. Lin and S. Saripalli, "Sampling-based path planning for UAV collision avoidance," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 11, pp. 3179–3192, 2017.

[7] H. Surden and M. A. Williams, "Technological opacity, predictability, and self-driving cars," *Cardozo L. Rev*, vol. 38, pp. 121–128, 2016.

[8] R. P. D. Vivacqua, M. Bertozzi, P. Cerri and R. F. Vassallo, "Self-localization based on visual lane marking maps: An accurate low-cost approach for autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 2, pp. 582–597, 2017.

[9] S. Liu, L. Li, J. Tang, S. Wu and J. L. Gaudiot, "Creating autonomous vehicle systems," *Synthesis Lectures on Computer Science*, vol. 6, no. 1, pp. 176–186, 2017.

[10] J. Wang, J. Liu and N. Kato, "Networking and communications in autonomous driving: A survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1243–1274, 2018.

[11] F. Santoso, M. A. Garratt and S. G. Anavatti, "Visual–inertial navigation systems for aerial robotics: Sensor fusion and technology," *IEEE Transactions on Automation Science and Engineering*, vol. 14, no. 1, pp. 260–275, 2016.

[12] S. Gupta, J. Davidson, S. Levine, R. Sukthankar and J. Malik, "Cognitive mapping and planning for visual navigation," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, pp. 2616–2625, 2017.

[13] C. Tang, Z. Xu and M. Tomizuka, "Disturbance-observer-based tracking controller for neural network driving policy transfer," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 9, pp. 3961–3972, 2019.

[14] M. S. Mahdavinejad, M. Rezvan, M. Barekatain, P. Adibi, P. Barnaghi *et al.,* "Machine learning for internet of things data analysis: A survey," *Digital Communications and Networks*, vol. 4, no. 3, pp. 161–175, 2018.

[15] V. A. Laurense, J. Y. Goh and J. C. Gerdes, "Path-tracking for autonomous vehicles at the limit of friction," in *Proc. American Control Conf.*, Seattle, USA, pp. 5586–5591, 2017.

[16] W. Shi, M. B. Alawieh, X. Li and H. Yu, "Algorithm and hardware implementation for visual perception system in autonomous vehicle: A survey," *Integration*, vol. 59, pp. 148–156, 2017.

[17] M. Zakarya and L. Gillam, "Energy efficient computing, clusters, grids and clouds: A taxonomy and survey," *Sustainable Computing: Informatics and Systems*, vol. 14, pp. 13–33, 2017.

[18] R. Anil, G. Pereyra, A. Passos, R. Ormandi, G. E. Dahl *et al.,* "Large scale distributed neural network training through online distillation," in *Proc. Int. Conf. on Machine Learning*, New York, United States, pp. 1–8, 2018.

[19] J. Gowrishankar, T. Narmadha, M. Ramkumar and N. Yuvaraj, "Convolutional neural network classification on 2d craniofacial images," *International Journal of Grid and Distributed Computing*, vol. 13, no. 1, pp. 1026–1032, 2020.

[20] G. Ciaparrone, F. L. Sanchez, S. Tabik, L. Troiano, R. Tagliaferri *et al.,* "Deep learning in video multi-object tracking: A survey," *Neurocomputing*, vol. 381, pp. 61–88, 2020.

[21] S. Brilly Sangeetha, N. R. Wilfred Blessing, N. Yuvaraj and J. Adeline Sneha, "Improving the training pattern in back-propagation neural networks using holt-winters' seasonal method and gradient boosting model," In: P. Johri, J. Verma and S. Paul (Eds.) *Applications of Machine Learning. Algorithms for Intelligent Systems*, Singapore: Springer, pp. 189–198, 2020.

[22] Z. Q. Zhao, P. Zheng and X. Wu, "Object detection with deep learning: A review," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 11, pp. 3212–3232, 2019.

[23] Y. Li, X. Zhang, H. Li and Z. Xiao, "Object detection and tracking under complex environment using deep learning-based LPM," *IET Computer Vision*, vol. 13, no. 2, pp. 157–164, 2018.

[24] Y. Yan, J. Ren, H. Zhao, G. Sun and J. Soraghan, "Cognitive fusion of thermal and visible imagery for effective detection and tracking of pedestrians in videos," *Cognitive Computation*, vol. 10, no. 1, pp. 94–104, 2018.

[25] A. Brunetti, D. Buongiorno, G. F. Trotta and V. Bevilacqua, "Computer vision and deep learning techniques for pedestrian detection and tracking: A survey," *Neurocomputing*, vol. 300, pp. 17–33, 2018.

[26] J. E. Van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Machine Learning*, vol. 109, no. 2, pp. 373–440, 2020.

[27] M. Mahmud, M. S. Kaiser, A. Hussain and S. Vassanelli, "Applications of deep learning and reinforcement learning to biological data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 6, pp. 2063–2079, 2018.

[28] D. Zhang, X. Han and C. Deng, "Review on the research and practice of deep learning and reinforcement learning in smart grids," *CSEE Journal of Power and Energy Systems*, vol. 4, no. 3, pp. 362–370, 2018.

[29] A. Nagabandi, G. Kahn, R. S. Fearing and S. Levine, "Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning," in *Proc. IEEE Int. Conf. on Robotics and Automation*, Brisbane, QLD., Australia, pp. 7559–7566, 2018.

[30] Fisher Yu, "BDD100K: A large-scale diverse driving video database," Available: https://bair.berkeley.edu/blog/2018/05/30/bdd/, Accessed on 15.03.2020, 2018.

[31] F. Diba and E. Esmailzadeh, "Components sizing optimisation of hybrid electric heavy duty truck using multi-objective genetic algorithm," *International Journal of Heavy Vehicle Systems*, vol. 27, no. 3, pp. 387–404, 2020.

[32] G. Bureika, M. Levinzon, S. Dailydka, S. Steisunas and R. Zygiene, "Evaluation criteria of wheel/rail interaction measurement results by trackside control equipment," *International Journal of Heavy Vehicle Systems*, vol. 26, no. 6, pp. 747–764, 2019.

[33] M. Chai, S. Rakheja and W. B. Shangguan, "Relative ride performance analysis of a torsio-elastic suspension applied to front, rear and both axles of an off-road vehicle," *International Journal of Heavy Vehicle Systems*, vol. 26, no. 6, pp. 765–789, 2019.

[34] X. R. Zhang, W. F. Zhang, W. Sun, X. M. Sun and S. K. Jha, "A robust 3-D medical watermarking based on wavelet transform for data protection," *Computer Systems Science & Engineering*, vol. 41, no. 3, pp. 1043–1056, 2022.

[35] X. R. Zhang, X. Sun, X. M. Sun, W. Sun and S. K. Jha, "Robust reversible audio watermarking scheme for telemedicine and privacy protection," *Computers, Materials & Continua*, vol. 71, no. 2, pp. 3035–3050, 2022.

[36] N. Yuvaraj, K. Praghash, R. A. Raja and T. Karthikeyan, "An investigation of garbage disposal electric vehicles (GDEVs) integrated with deep neural networking (DNN) and intelligent transportation system (ITS) in smart city management system (SCMS)," *Wireless Personal Communications*, vol. 123, pp. 1–20, 2021. https://doi.org/10.1007/s11277-021-09210-8.

[37] X. Li, Z. Xie and Y. Pi, "Traffic sign detection based on improved faster R-CNN for autonomous driving," *The Journal of Supercomputing*, vol. 89, pp. 1–21, 2022.

[38] T. Turay and T. Vladimirova, "Towards performing image classification and object detection with convolutional neural networks in autonomous driving systems: A survey," *IEEE Access*, vol. 10, pp. 14076–14119, 2022.

[39] G. Babu Naik, P. Ameta, N. Baba Shayeer and S. Kavya Dravida, "Convolutional neural network based on self-driving autonomous vehicle (CNN)," *Proc. Innovative Data Communication Technologies and Application*, Singapore, pp. 929–943, 2022.